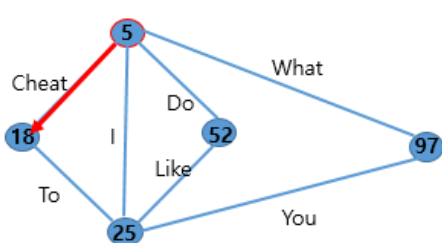


# WEEK14

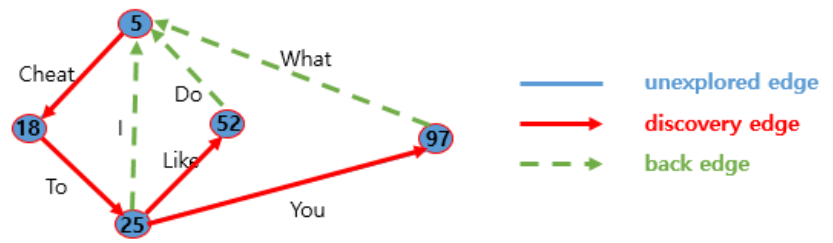
- 주의 사항: 부정행위 금지(채점서버 외 인터넷 사용금지), STL 사용금지 (string, vector는 사용 가능)
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

## 문제 1 (Hard version)

데이터를 입력 받아 인접리스트(Adjacent List) 기반의 무향그래프를 생성하고 깊이 우선 탐색(DFS) 를 이용하여 모든 정점과 간선을 1번씩 탐색 하는 프로그램을 만들어 보자. 아래의 그래프는 2page의 예제입력과 같은 순서로 생성되었다.



5번 정점에서 DFS 시작 상태



DFS 종료 후 모든 정점, 간선에 대한 처리 완료 상태

데이터를 입력 받아 그래프에 정점과 간선을 삽입한 뒤, DFS를 수행한다. DFS 수행간 임의의 정점 A에서 방문할 수 있는 정점이 여러 개 있다면 그래프가 생성될 때, 정점 A와 먼저 연결되었던(간선을 생성한) 정점을 우선적으로 탐색한다. 간선은 그래프에 존재하는 두 정점으로 주어진다. 만약 두 정점간에 이미 간선이 있을 경우, -1을 출력한다. 시작점이 주어지고 해당 시작점으로부터 DFS를 수행하며 방문한 정점 순서대로 해당 정점들의 정보를 출력한다.

**Hard:** 추가적으로 탐색한 모든 간선들에 대하여 Discovery Edge인지 Back Edge인지 구분하려 한다.

따라서 Discovery Edge들의 정보를 출력하고 Back Edge들의 정보를 그 다음 줄에 출력한다. 이 때, 간선의 종류가 같은 경우, 먼저 만들어진 간선의 정보가 먼저 출력된다.

# 입력

첫째 줄에는 정점의 개수  $N$  ( $1 \leq N \leq 1,000$ )과 간선의 개수  $M$  ( $1 \leq M \leq 100,000$ )와 DFS 시작 정점 번호  $A$ 가 주어진다.

둘째 줄에는 그래프에 삽입될  $N$ 개의 정점번호  $I$  ( $1 \leq I \leq 20,000$ )가 공백으로 구분되어 입력된다.

이후  $M$ 개의 줄을 통해 간선 정보로 두 정점  $S, D$  ( $1 \leq S, D \leq 20,000$ )와 간선에 저장될 문자열  $U$ 가 주어진다.

# 출력

DFS 수행 완료 후, 첫 번째 줄에는 정점들의 정보를 방문한 순서대로 출력한다.

두 번째 줄에는 Discovery Edge들 중 먼저 삽입된 간선의 정보부터 출력한다.

세 번째 줄에는 Back Edge들 중 먼저 삽입된 간선의 정보부터 출력한다.

## 예제 입출력 1

예제 입력	예제 출력
5 8 5	-1
5 18 25 52 97	5 18 25 52 97
5 18 Cheat	Cheat To Like You
5 25 I	I Do What
5 52 Do	
18 5 Hobby	
5 97 What	
18 25 To	
25 52 Like	
25 97 You	